

Alphreschools Assignment > 1

Question No-1

Input = 1253

```
#include <iostream>
using namespace std;
```

```
int main()
{
    int n, r, sum = 0, temp;
    cout << "Enter the number = ";
    cin >> n;
    temp = n;
    while (n > 0)
    {
        r = n % 10;
        sum = sum + (r * r * r);
        n = n / 10;
    }
    if (temp == sum)
        cout << "Armstrong Number" << endl;
    else
        cout << "Not Armstrong Number" << endl;
    return 0;
}
```

Output →

$$1253 \Rightarrow 1*1*1 + 2*2*2 + 5*5*5 + 3*3*3$$

$$\Rightarrow 1 + 8 + 125 + 27 \Rightarrow 161$$

Not a Armstrong number

Question No 2

```
#include <bits/stdc++.h>
using namespace std;
```

```
int first (int arr[], int low, int high, int x, int n)
{
    if (high >= low) {
        int mid = low + (high - low) / 2;
        if ((mid == 0 || x > arr[mid-1]) && arr[mid] == x)
            return mid;
        else if (x > arr[mid])
            return first(arr, low, (mid-1), x, n);
    }
}
```

2

```
return -1;
```

2

```
int main()
```

```
{
```

```
int arr[] = {1, 3, 5, 5, 5, 5, 7, 12, 3, 12, 5};
```

```
int n = sizeof(arr) / sizeof(int);
```

```
int x = 7;
```

```
findFirstAndLast(arr, n, x);
```

```
return 0;
```

2

Output - first occurrence = 5

last occurrence = 7

Question No-3

```
#include <iostream>
using namespace std;
```

```
main() {
```

```
    // rows
```

```
    for (int i = 0; i < 5; i++) {
```

```
        // columns
```

```
        for (int j = 0; j < 5; j++) {
```

```
            cout << "X" << " ";
```

```
        }
```

```
        cout << endl;
```

```
    }
```

```
}
```

```

X  X  X  X  X
X  X  X  X  X
X  X  X  X  X
X  X  X  X  X
X  X  X  X  X
```

Output

Question No - 25

```
#include <iostream>
#include <stdio.h>
using namespace std;
```

```
void toggle case (char *str);
```

```
int main ()
```

```
{
    char str [MAXSIZE]
```

```
{
    while (*str)
```

```
{
    if (*str >= 'a' && *str <= 'z')
```

```
        *str = *str - 32;
```

```
    else if (*str >= 'A' && *str <= 'Z')
```

```
        *str = *str + 32;
```

```
    str++;
```

```
}
```

```
}
```

Input → Programmer

String after the character are

Output → PROGRAMMER